WHAT IS CLAIMED IS:

1. A method of storing a texel in a texel cache comprising:

reading a t coordinate of the texel, the t coordinate comprising a plurality of bits;

reading a s coordinate of the texel, the s coordinate comprising a plurality of bits; and

forming an offset by concatenating bits of the t coordinate with bits of the s coordinate.

2. The method of claim 1 further comprising forming an index by concatenating bits of the t coordinate with bits of the s coordinate.

3. The method of claim 2 further comprising storing the texel in a texel cache comprising a plurality of cache lines, wherein each cache line comprises a plurality of storage elements.

4. The method of claim 3 further comprising storing the texel in a storage element identified by the offset.

5. The method of claim 4 further comprising storing the texel in a cache line identified by the index.

6. The method of claim 5 wherein the forming an offset by concatenating bits of the t coordinate with bits of the s coordinate is done by concatenating the lower bits of the t coordinate with the lower order bits of the s coordinate.

7. The method of claim 6 further comprising retrieving the texel from a main memory, wherein the texel has an address in main memory.

8. The method of claim 7 further comprising forming a tag by concatenating the high order bits of the s coordinate and high order bits of the t coordinate, adding the address in main memory, and storing the tag in a look-up table.

9. The method of claim 8 wherein the forming an index by concatenating bits of the t coordinate with bits of the s coordinate is done by

concatenating middle order bits of the t coordinate with middle order bits of the s coordinate.

10. The method of claim 1 wherein the texel is associated with a MIPmap having a level of detail comprising a plurality of bits, further comprising forming an index signal by concatenating middle order bits of the s coordinate, middle level bits of the t coordinate, and at least one bit of the level of detail.

11. The method of claim 1 wherein the texel is associated with a texture having a texture identification comprising a plurality of bits, further comprising forming an index signal by concatenating middle order bits of the s coordinate, middle level bits of the t coordinate, and at least one bit of the texture identification.

12. The method of claim 1 wherein the texel is associated with a texture having an r coordinate comprising at least one bit, further comprising forming an index signal by concatenating middle order bits of the s coordinate, middle level bits of the t coordinate, and at least one bit of the r coordinate.

13. The method of claim 1 wherein the texel has a main memory address comprising a plurality of bits, further comprising forming an index signal by concatenating middle order bits of the s coordinate, middle level bits of the t coordinate, and at least one bit of the main memory address.

14. An integrated circuit comprising:

a texture cache subsystem for storing a texel;

a cache address generator subsystem for receiving an s and a t coordinate, and for providing an index and offset to the texture cache subsystem; and

a graphics pipeline subsystem for providing the s and t coordinates to the cache address generator subsystem, and memory addresses to the cache address generator subsystem, and further for receiving a texel from the texture cache subsystem,

wherein the index and offset comprise bits of the s and t coordinates.

15. The integrated circuit of claim 14 wherein the index further comprises at least one bit of a level of detail signal associated with the texel.

16. The integrated circuit of claim 14 wherein the index further comprises at least one bit of a texture id associated with the texel.

17. The integrated circuit of claim 14 wherein the index further comprises at least one bit of a memory address associated with the texel.

18. The integrated circuit of claim 14 wherein the index further comprises at least one bit of an r coordinate associated with the texel.

19. A method of generating a texture cache address comprising:

reading an s coordinate of a texture, wherein the s coordinate comprises a plurality of bits;

passing lower order bits of the s coordinate as a portion of an offset;

passing middle order bits of the s coordinate as a portion of an index; and

passing high order bits of the s coordinate as a portion of a tag.

20. The method of claim 19 further comprising:

reading a t coordinate of the texture, wherein the t coordinate comprises a plurality of bits;

passing lower order bits of the t coordinate as a portion of an offset;

passing middle order bits of the t coordinate as a portion of an index; and

passing high order bits of the t coordinate as a portion of a tag.

21. The method of claim 20 further comprising reading a memory address, and passing a portion of the memory address as a portion of the tag.

22. The method of claim 21 further comprising reading a level of detail associated with the texel, and passing a portion of the level of detail as a portion of the index.

23. The method of claim 21 further comprising reading a texture id associated with the texel, and passing a portion of the texture id as a portion of the index.

24. The method of claim 21 further comprising reading a memory address associated with the texel, and passing a portion of the memory address as a portion of the index.

25. The method of claim 21 further comprising reading an r coordinate associated with the texel, and passing a portion of the r coordinate as a portion of the index.

26. A method of storing address information for a group of texels comprising:

storing an index and offset of a first texel; and

storing a line index of at least one of the other quad of texels,

wherein the stored address information comprises information necessary to determine a group of addresses in a texture cache.

27. The method of claim 26 wherein the texture cache comprises a plurality of cache lines, further comprising storing an indication of which cache lines need to be updated.

28. The method of claim 27 further comprising storing information as to the wrap mode used to store the group of texels in the texel cache.

29. A graphics processor comprising:

a graphics pipeline coupled to receive texels and provide s and t coordinates and a memory address;

a cache address generator, coupled to receive the s and t coordinates and the memory address, and configured to generate an offset, an index, and a tag using the s and t coordinates and the memory address, and

a cache unit, coupled to provide texels, and receive the offset, index and tag, wherein the cache unit further comprises a texture cache.

30. A computer system comprising:

a central processing unit (CPU);

a main memory coupled to the CPU; and

a graphics processor as set forth in claim 29, coupled to the CPU.

31. A graphics processor comprising:

a texture cache manager, coupled to receive texel addresses and provide a packet of data, and a fetch request;

a memory controller coupled to receive the fetch request and provide a cache line update;

a FIFO coupled to receive and provide packets, and configured to store a plurality of packets; and

a texture cache controller coupled to receive data stored in cache lines,

wherein the packets comprise an offset of a first texel, an index of a first texel, and information required to generate an offset and an index of a second texel.

32. A computer system comprising:

a central processing unit (CPU);

a main memory coupled to the CPU; and

a graphics processor as set forth in claim 31, coupled to the CPU.